

1 1. A method comprising:
2 writing a dirty cache line to a disk drive prior
3 to a disk driver loading; and
4 monitoring for a disk write request prior to said
5 disk driver loading.

1 2. The method of claim 1 further comprising logging
2 said disk write request if said disk driver has not loaded.

1 3. The method of claim 2 further comprising
2 executing said disk write request.

1 4. The method of claim 1 further comprising
2 refreshing said cache line after said disk driver is
3 loaded.

1 5. The method of claim 1 further comprising
2 monitoring for said write requests by executing code that
3 is stored in a second memory.

1 6. The method of claim 1 further comprising
2 monitoring for said write requests by executing code that
3 provides an interface between a basic input output system
4 and a read only memory.

1 7. The method of claim 6 further comprising
2 acknowledging a filter function.

1 8. The method of claim 7 further comprising
2 executing said second memory code to record information of
3 a write to said disk drive.

1 9. The method of claim 8 further comprising
2 executing said basic input output system code to execute
3 said write requests.

1 10. The method of claim 1 further comprising
2 monitoring for said write requests by executing code that
3 modifies a stack.

1 11. The method of claim 10 further comprising
2 determining a stack offset.

1 12. The method of claim 11 further comprising using
2 said stack offset to return control to an option read only
3 memory.

1 13. An article comprising a medium storing
2 instructions, that if executed, enable a processor-based
3 system to:
4 write a dirty cache line to a disk prior to a
5 disk driver loading; and
6 monitor for a write request, prior to said disk
7 driver loading.

1 14. The article of claim 13 further storing
2 instructions, that if executed, enable a processor-based
3 system to log said write request if said disk driver has
4 not loaded.

1 15. The article of claim 14 further storing
2 instructions, that if executed, enable a processor-based
3 system to execute said write request to said disk.

1 16. The article of claim 13 further storing
2 instructions, that if executed, enable a processor-based
3 system to refresh said cache line if said disk driver is
4 loaded.

1 17. The article of claim 13 further storing
2 instructions, that if executed, enable a processor-based
3 system to monitor for said write request by executing code
4 that is stored in an option read only memory.

1 18. The article of claim 13 further storing
2 instructions, that if executed, enable a processor-based
3 system to monitor for said write request by executing code
4 that provides and interface for a basic input output system
5 and an option read only memory.

1 19. The article of claim 13 further storing
2 instructions, that if executed, enable a processor-based

3 system to monitor for said write request by executing code
4 that modifies a stack.

1 20. A system comprising:
2 a processor;
3 a disk drive coupled to said processor;
4 a disk cache coupled to said processor and said
5 disk drive; and
6 at least one memory device coupled to said
7 processor storing instructions that, if executed, enable
8 said system to write a dirty cache line to said disk drive
9 prior to loading a disk driver, and to monitor for a disk
10 write request prior to loading said disk driver.

1 21. The system of claim 20 wherein said at least one
2 memory device stores instructions, that if executed, enable
3 said system to log said disk write request if said disk
4 driver has not loaded.

1 22. The system of claim 21 wherein said at least one
2 memory device stores instructions, that if executed, enable
3 said system to execute said disk write request.

1 23. The system of claim 20 wherein said at least one
2 memory device stores instructions, that if executed, enable
3 said system to a refresh cache line after said disk driver
4 is loaded.

1 24. The system of claim 20 wherein said at least one
2 memory device stores instructions, that if executed, enable
3 said system to monitor for said write requests by executing
4 code that is stored in an option read only memory.

1 25. The system of claim 20 wherein said disk cache
2 comprises a polymer memory.

1 26. The system of claim 20 wherein said disk cache
2 comprises ferroelectric polymer memory.

1 27. A method comprising acknowledging a filter
2 function for a second memory.

1 28. The method of claim 27 further comprising sending
2 disk drive identification data to code executing from said
3 second memory.

1 29. The method of claim 27 further comprising
2 executing code from said second memory to write to a disk
3 drive.

1 30. The method of claim 27 further comprising
2 initializing a second memory as a drive request handler.

1 31. The method of claim 27 wherein said second memory
2 further comprises an option read only memory.

1 32. The method of claim 30 further comprising
2 determining a stack offset.

1 33. The method of claim 30 further comprising using
2 said stack offset to return control to said second memory.